

DDS: A new Dynamic Distribution System

A. Manafov¹, A. Lebedev¹, and M. Al-Turany¹

¹GSI, Darmstadt, Germany

Dynamic Distribution System (DDS) is a tool-set that automates and dramatically simplifies the process of distribution of user defined processes with their dependencies on any resource management system using a given topology.

DDS is a successor of PoD [1][2]. Unlike PoD, which automates PROOF [3] deployment, DDS will handle any kind of user processes with complex dependencies between processes. The system is designed and being implemented within the new ALFA framework [4].

Concept

During 2013 a conceptual design of the system has been developed.

A key point of this design is the so called topology language, i.e.: DDS is a user oriented system the definition of the topology by the user has to be simple and powerful at the same time. The basic building block of the system is a task. Namely, a task is a user defined executable or a shell script, which will be deployed and executed by DDS on a Resource Management System. To describe dependencies between tasks in a topology we use properties. For example, if one task wants to communicate with another task they can have the same property of a type TCP/IP port. In this case DDS will notice that, will find out a free port number on a destination system and set this number to configuration files of both user processes. Moreover, there will be different types of properties, for example, tasks can be dependent on each other via a file or a named pipe.

Tasks can be grouped into DDS collections and DDS groups. The difference between collections and groups is that collections are a signal to DDS topology parser that tasks of given collections will be executed on the same physical machine. This is useful if tasks have a lot of communication or they want to access the same shared memory. A set of tasks and task collections can be also grouped into task groups.

DDS utilizes a plug-in system in order to use different job submission front-ends. The first and the main plug-in of the system will be an SSH plug-in, which can be used to dynamically turn a bunch of machines to user worker nodes. The SSH plug-in is a perfect tool for a Cloud based solutions.

Outlook

The DDS system is in the initial development phase. We expect a lot of new features in the upcoming year.

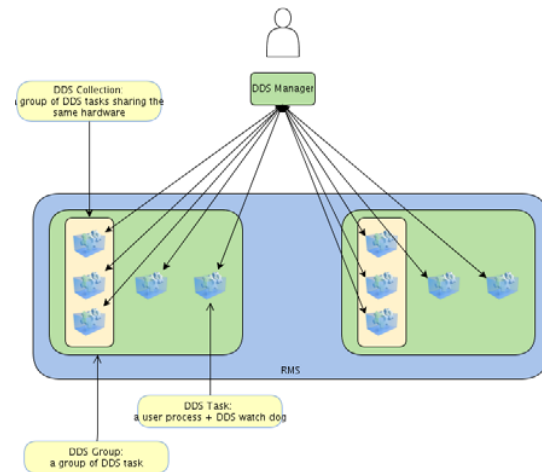


Figure 1: Illustration of the DDS concept.

1. Operational prototype of DDS with the ability to parse topologies such as examples of FairROOT tasks.
2. The operational prototype will be able to deploy single dependency tasks like a PROOF cluster or FairROOT analysis tasks. The deployment will be steered using the SSH plug-in on the Cloud and local computers.
3. Extend the plug-in machinery to cover the following Resource Management Systems: SLURM, LSF, Torque (PBS), Grid Engine, Condor, Panda.
4. Extend the submission machinery to be able to deploy multi-dependency tasks.
5. Extend the complexity of the topology language to support custom types of task properties.

References

- [1] A. Manafov et al, "PROOF on Demand", IT-07, GSI Scientific Report 2012.
- [2] PROOFonDemand(PoD), <http://pod.gsi.de>
- [3] TheParallelROOTFacility(PROOF), <http://root.cern.ch/drupal/content/proof>
- [4] M. Al-Turany et al., Status of the FairRoot framework, this report.